11. Sean

Program Name: Sean.java **Test Input File: sean.dat**

Sean loves mazes, and has noticed while mazes are solved, rarely do you get to see what the maze looks like afterwards. So here is his maze for you to solve. Create a maze and then give the location of every position away from indicated starting points. Oh, by the way, there can be multiple starting points. Well shoot, the maze isn't even necessarily a rectangle. Good news, you are given the number of rows, but each row can be a different number of columns. In fact, each maze can have multiple mazes inside it, and there might not even be a starting point in that maze. At least the entire maze is surrounded by a capital X.

Input: One dataset with the first item an integer N indicating the number of rows in the maze. This will be followed by a design of the maze outlined by "X" characters. After the maze there will be an unknown number of ordered pair starting locations. Each of these starting locations will consist of two integers, the first is the row and the second is the column. The starting point will not be on an X. **Note:** position (0,0) indicates the upper left-hand corner of the maze.

Output: The maze with each available spot displaying the shortest distance from any starting spot. Note: Each spot will only display the one's digit of the distance.

Sample Input:

```
15
XXXXXXXXXXXXXXX
X          X   X
X              X
X   X          X
X              X
X   X          XXXXXXX
X       X             X
X       X         XXXX
X       X     XXXX  X
X       X     X     X
XXX  XXXXX    XXXXXXX
  X      X        X X
  X      X    XXXXX X
  X           X     X
  XXXXXXXXXXXXXXXXXXX
3 3
6 3
7 9
8 19
```

Sample Output:

```
XXXXXXXXXXXXXXX
X222223456X666X
X2111234555555X
X210X234444444X
X2111233333334X
X211X233222234XXXXXXX
X2101233X11234567890X
X2111234X01234567XXXX
X2222234X11234XXXX10X
X3333334X22234X43211X
XXX44XXXXX3334XXXXXXX
  X555678X44445678X X
  X666677X5555XXXXX X
  X77777766666X     X
  XXXXXXXXXXXXXXXXXXX
```